



# Une Méthode Hybride pour le Problème de Tournées de Véhicules avec Contraintes de Temps

Sohaib Afifi, Aziz Moukrim

## ► To cite this version:

Sohaib Afifi, Aziz Moukrim. Une Méthode Hybride pour le Problème de Tournées de Véhicules avec Contraintes de Temps. ROADEF 2016, 17ème congrès annuel de la Société française de Recherche Opérationnelle et d'Aide à la Décision, Feb 2016, Compiègne, France. hal-01302053

**HAL Id: hal-01302053**

**<https://hal.science/hal-01302053>**

Submitted on 13 Apr 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Une Méthode Hybride pour le Problème de Tournées de Véhicules avec Contraintes de Temps

Sohaib Afifi<sup>1</sup>, Aziz Moukrim<sup>2</sup>

<sup>1</sup> Université d'Artois, LGI2A, F-62400 Béthune, France  
`sohaib.afifi@univ-artois.fr`

<sup>2</sup> Sorbonne universités, Université de technologie de Compiègne  
CNRS, Heudiasyc UMR 7253, CS 60 319, 60 203 Compiègne cedex  
`aziz.moukrim@hds.utc.fr`

**Mots-clés :** *méthodes hybrides, tournées de véhicules, fenêtres de temps*

## Formulation du problème

Ce travail propose une nouvelle méthode de résolution pour les problèmes de tournées de véhicules (VRP). Le but est de servir un ensemble de clients avec un ensemble de véhicules tout en respectant certaines contraintes. Le VRPTW (vehicle routing problem with time windows) est la variante du VRP la plus étudiée. Des fenêtres de temps de visites sont associées aux clients et au dépôt. Chaque arc étant dans ce contexte caractérisé par une durée de trajet généralement assimilée à un coût. La disponibilité d'un client  $i$  est représentée par une fenêtre de temps définie par une date de service au plus tôt  $e_i$  et une date de service au plus tard  $l_i$ . Un véhicule se rendant au client  $i$  plus tôt que  $e_i$  doit attendre jusqu'au début de la fenêtre de temps. Le temps d'attente résultant est pris en compte dans la contrainte de durée maximum de route. Une arrivée tardive après  $l_i$  rend la solution associée irréalisable.

De nombreuses formulations et méthodes exactes sont proposées pour le VRPTW. Nous pouvons citer quelques méthodes récentes : le branch-and-cut [1], le branch-and-price et le branch-and-price-and-cut [2]. Les métaheuristiques proposées pour le VRPTW sont les algorithmes de recherche tabu, le recuit simulé et les algorithmes génétiques [3].

## Notre approche

Dans ce travail, nous proposons dans un premier temps une nouvelle formulation linéaire pour le VRPTW. Elle utilise  $O(n^2)$  variables binaires et  $O(n \times m^2)$  contraintes. La résolution de cette formulation est par la suite accélérée en réduisant à la fois la taille de l'arbre du branch and bound et en découvrant les solutions faisables plus tôt pendant l'exécution.

Les heuristiques primitives sont un aspect très important dans la résolution des programmes linéaires en nombres entiers. Nous proposons et testons un nombre d'heuristiques dédiées à nos problèmes et discutons leur intégration dans l'algorithme de branch-and-bound. Nous comptons trois types d'heuristiques primitives : les heuristiques d'initialisation, les heuristiques plongeantes et la recherche locale.

Une autre stratégie est de réduire l'exploration de l'arbre du Branch-and-bound en vérifiant la faisabilité des nœuds à l'aide des tests de faisabilité qui exploitent la spécification du problème. Dans ce sens, nous proposons l'utilisation du raisonnement énergétique et les méthodes basées sur les cliques et des relaxations à des problèmes de bin-packing [4].

## Résultats numériques

Nous avons testé notre méthode sur les instances de Solomon. L'algorithme est codé en C++ en utilisant la bibliothèque SCIP et CPLEX et est compilé avec le GCC de GNU dans un

environnement Linux. L'algorithme est par la suite comparé avec le même solveur sans ajouter nos composants d'accélération. Un temps limite d'une heure est imposé pour chaque exécution.

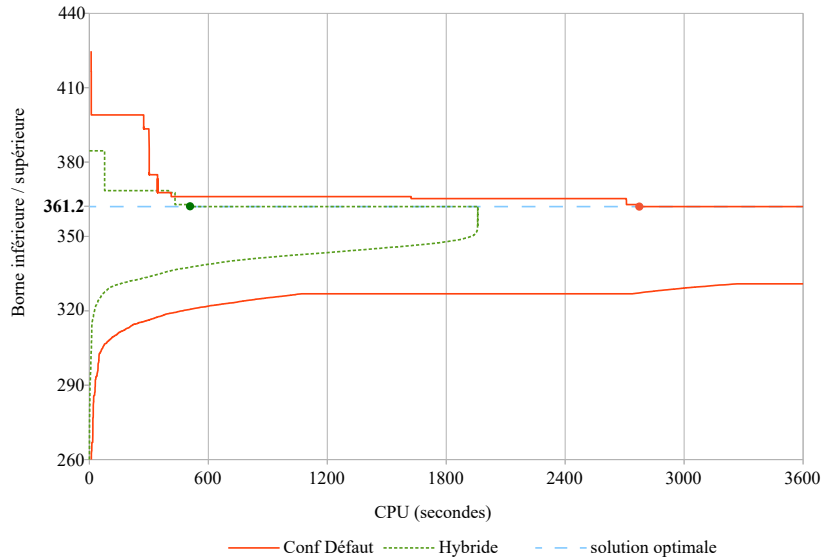


FIG. 1 – Développement des bornes pour une instance de taille moyenne avec la configuration par défaut (ligne continue) et notre algorithme (ligne en pointillé)

Les heuristiques utilisées sont capables de générer rapidement des solutions faisables. Elles réduisent considérablement le temps de calcul tout en améliorant la qualité des solutions. D'autre part, la vérification de faisabilité et les coupes ajoutées peuvent prédire l'infaisabilité des branches parcourues dans des niveaux plus tôt et par conséquent diminuent le temps utilisé pour améliorer la borne inférieure. Fig.1 présente un exemple sur une instance de taille moyenne et montre comment l'introduction de ces techniques a considérablement réduit le temps cpu sur les deux côtés, la borne supérieure et la borne inférieure. Par ailleurs, l'algorithme implique un grand nombre de paramètres. Une première piste qui nous semble logique est de lancer, par exemple, les heuristiques plus tôt au niveau de l'arbre et les tests de faisabilité sur les nœuds avec des graphes de conflits denses.

**Remerciements** Ce travail est soutenu par l'ANR (Agence Nationale de la Recherche) dans le cadre du projet TCDU (Transport Collaboratif dans la Distribution Urbaine ANR-14-CE22-0017). Il est également effectué dans le cadre du Labex MS2T soutenu par le gouvernement français, à travers le programme "Investissements d'avenir" géré par l'Agence Nationale de la Recherche (Référence ANR-11-IDEX-0004-02).

## Références

- [1] Jonathan F Bard, George Kontoravdis, and Gang Yu. A branch-and-cut procedure for the vehicle routing problem with time windows. *Transportation Science*, 36(2) :250–269, 2002.
- [2] Mads Jepsen, Bjørn Petersen, Simon Spoorendonk, and David Pisinger. Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research*, 56(2) :497–511, 2008.
- [3] Thibaut Vidal, Teodor Gabriel Crainic, Michel Gendreau, and Christian Prins. A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers & Operations Research*, 40(1) :475–489, 2013.
- [4] Sohaib Afifi, Rym Nesrine Guibadj, and Aziz Moukrim. New lower bounds on the number of vehicles for the vehicle routing problem with time windows. In *CPAIOR 2014, Cork, Ireland*, volume 8451 of *Lecture Notes in Computer Science*, pages 422–437. Springer, 2014.